REMARKS

Applicant appreciates the time taken by the Examiner to review Applicant's present application. Applicant has added Claims 36-38. Applicant submits that no new matter has been added. Thus, Claims 1-38 remain pending in this application. This application has been carefully reviewed in light of the Official Action mailed January 3, 2006. Applicant respectfully requests reconsideration and favorable action in this case.

Rejections under 35 U.S.C. § 112

Claims 5 and 22 stand rejected under 35 U.S.C. § 112 second paragraph. The Examiner asserts that the limitation "the request" in these claims has insufficient antecedent basis. Applicant respectfully disagrees. Claim 5 and 22 indirectly depend, respectively, on Claims 1 and 18. Claims 1 and 18 both recite, in part: "evaluating a parameter of a request". Thus, Applicant believes there is sufficient antecedent basis for the limitation "the request" and respectfully requests the withdrawal of this rejection.

Rejections under 35 U.S.C. § 102

Claims 1-9, 15-17, 18-26 and 32-34 stand rejected as anticipated by U.S. Patent No. 5,946,697 ("Shen"). Applicant respectfully traverses this rejection.

Overview of the Invention

Claim 1 of the application recites a method for caching, comprising registering a module, evaluating a parameter of a request, wherein the parameter is evaluated by the module, creating a signature based on the evaluation, searching for responsive content in a cache based on the signature and generating responsive content and storing it in the cache if no responsive content is found in the cache.

Thus, embodiments of the present invention may provide a method for caching which may be usefully employed at a web server. A request for content may be sent from a client computer and received at a web server. When the request is received at the web server the caching framework may utilize one or more modules to evaluate the parameters of the request (e.g. one module may evaluate or process a parameter of the request). Based on the evaluation of each of the parameters (accomplished by each of the modules) a signature (e.g.

corresponding to the requested content) may be created. This signature can be used to search for the requested content in the cache. If the requested content is not in the cache, the requested content may be generated (e.g. by an application server), stored in the cache at the web server and provided to the requestor at the client computer.

The caching framework provided by embodiments of the present invention may also store a variety of metadata associated with the content in the cache. This metadata may comprise, for example, template metadata pertaining to the content (e.g. which template a piece of content is associated with), or request metadata pertaining to the content (e.g. metadata pertaining to the request which generated the content, etc. Request metadata for storing in association with the content may be provided to the caching framework by one or more of the same modules which are responsible for evaluating the parameters of the request. Thus, the request metadata stored with the content in the cache may pertain to the parameters of the request which originally generated the content.

Associating metadata with content in the cache at the web server allows this content to be regenerated and placed in the cache without receiving a new request (e.g. from a client computer). When content changes metadata regarding the changed content may be communicated to the caching framework. This metadata may be compared to the metadata associated with each piece of content in the cache. If a match is found it may indicate that the content in the cache is affected by the changed content. Thus, to update the content in the cache the request which resulted in that piece of content being generated originally may be regenerated from the metadata associated with that content. A response (e.g. content responsive to the regenerated request) can then be returned to the caching framework and used to replace the affected content.

Overview of Shen

Shen, in contrast, relates to improving the transfer rate of hierarchically structured files (e.g. HTML) over a network. (*See*, Shen Col. 1, Lines 1-8, Col. 2, Lines 25-40)

More specifically, a user at a client computer may request a site. It may be determined if the HTML file for the site has been cached, and if it has not been cached the request may be sent to a web server, an HTML file responsive to the request generated at the web server, the HTML file sent to the client computer and the HTML file caches at the client computer. (*See*, Shen FIG 3, Col. 6, Lines 11-40)

If, however, the HTML file corresponding to that request has been previously cached at the client computer an agent at the client computer extracts a macro name file and a macro definition file from the cached HTML file. (*See*, Shen FIG 3, Col. 6, Lines 55-60)

A macro name file is a file which comprises macro names formed from the cached HTML document. A macro name is a name that identifies a construct (i.e. list) in the cached HTML document. To form a micro name, Shen forms a checksum from the alphanumeric characters that comprise the construct or list. Thus, a macro name file comprises a set of checksums stored in the same sequence that that list or constructs to which they correspond appear in the cached HTML document. (*See*, Shen FIG 3, Col. 7, Lines 16-20, 22-25, 33-37, 42-46)

A macro definition file comprises a set of macro definitions, where each macro definition directly corresponds to a macro name and is simply the content of that portion of the body of the cached HTML file represented by that macro name. (*See*, Shen Col. 7, Lines 59-66) Thus, a macro definition file comprises a set of HTML constructs or lists.

After composing the macro name file and macro definition file from the cached HTML document the client agent append the macro name file to the request, and sends the request to the web server. (*See*, Shen FIG 3, Col. 8, Lines 3-8)

A server agent at the web server, fetches the HTML file referenced by the request and compresses this HTML file using the macro name file provided in the request. More specifically, the server agent compares the macro names of the HTML file with the macro name file received in the request, such that the compressed HTML file comprises the macro names for the portions of the HTML file stored on the server computer which have not changed interspersed with the full constructs or lists for those sections of the HTML file that have changed. (*See*, Shen FIG 3, Col. 8, Lines 26-35, 40-49, 54-61)

The compressed file is then returned to the client agent at the client computer which recovers the HTML file by expanding the compressed file using the macro definition file previously produced by the client agent to refresh the HTML document in the cache. (*See*, Shen FIG 3, Col. 9, Lines 10-20)

Discussion of Independent Claims 1 and 18

Claim 1 of the application recites a method for caching, comprising registering a module, evaluating a parameter of a request, wherein the parameter is evaluated by the module,

creating a signature based on the evaluation, searching for responsive content in a cache based on the signature and generating responsive content and storing it in the cache if no responsive content is found in the cache. Claim 18 recites similar limitations.

The Examiner states in the Office Action that "Shen teaches a method for registering a module (i.e. the macro definition file), evaluating a parameter (i.e. macro name file) of a request, wherein the parameter is evaluated by the module; creating a signature (i.e. the macro compressed file) based on the evaluation, searching for responsive content (i.e. changed HTML file content/portion) in a cache based on the signature and generating responsive content and storing it in the cache if no responsive content is found in the cache (i.e. transmitting the changed HTML file content from server to client location."

Thus, to encapsulate the Examiner's assertion it seems if the Examiner is equating:

the module of Claim 1 with the macro definition file of Shen;

the parameter of Claim 1 with the macro name file of Shen;

the signature of Claim 1 with the macro compressed file of Shen;

the responsive content of Claim 1 with changed HTML file content/portion of Shen; and

generating responsive content and storing it in the cache if no responsive content is found in the cache with the transmitting the changed HTML file content from server to client location as is done in Shen.

First and foremost, Applicant respectfully submits that a module which is designed to evaluate a parameter is not equivalent to a set of HTML constructs or lists (i.e. a macro definition file). Additionally, Claim 1 recites evaluating a parameter of a request, wherein the parameter of the request is evaluated by the module. The Examiner has equated the parameter of Claim 1 with the macro name file of Shen. Thus, translating this limitation of Claim 1 using the equivalent portions of Shen cited by the Examiner would mean that a set of checksums (i.e. the macro name file) would be evaluated by the a set of HTML constructs or lists (i.e. the macro definition file). The Applicant respectfully submits that the macro definition file (e.g. set of HTML constructs or lists) is not operable to evaluate a set of checksums. Moreover, in Shen only the macro name file is transmitted to the web server, while only the compressed file is transmitted from the server to the client. Thus, the macro name file and macro definition file of Shen are never used simultaneously by either the client computer or the web server of Shen. Accordingly, Applicant respectfully submits that Shen does not disclose at

least the limitations of registering a module or evaluating a request, wherein the parameter of the request is evaluated by the module.

Claim 1 also recites creating a signature based on the evaluation. For similar reasons as elaborated on above, Applicant respectfully submits that the compressed file is not created by based on the evaluation. Again, translating this limitations into the language of Shen cited by the Examiner would necessitate that the macro compressed file be created based on the evaluation of a set of checksums (i.e. the macro name file) by a set of HTML constructs or lists (i.e. the macro definition file). In addition to the above arguments regarding this evaluation, in Shen, the macro compressed file is created by comparing the macro name file with the HTML document at the server, not on the evaluation of the macro name file by the macro definition file. Accordingly, Applicant respectfully submits that Shen does not disclose at least the limitations of creating a signature based on the evaluation.

As Shen does not disclose all the limitations of Claim 1, Applicant respectfully requests the withdrawal of the rejection of Claim 1. Additionally, as Claim 18 recites limitations similar to Claim 1 the withdrawal of the rejection of Claim 18 is requested as well.

Dependent Claims 2-9, 15-17, 19-26 and 32-34

Dependent Claims 2-9, 15-17, 19-26 and 32-34 depend from either Claim 1 or 18. Consequently, Applicant respectfully submits that the above arguments apply equally well to these claims and accordingly requests the withdrawal of the rejection of Claims 2-9, 15-17, 19-26 and 32-34.

Rejections under 35 U.S.C. § 103

Claims 10-14, 27-31 and 35 stand rejected as obvious over U.S. Patent No. 5,946,697 ("Shen") in view of U.S. Publication No. 2002/0194219 ("Bradley"). As Claims 10-14, 27-31 and 35 depend indirectly from either Claims 1 or 18 Applicant submits that the foregoing arguments apply equally well to these claims and respectfully requests the withdrawal of their rejection.

Newly Added Claims 36-38

Applicant has added Claims 36-38 to distinctly point out and claim the invention. Applicant respectfully submits no new matter has been added. Applicant respectfully submits

that the cited prior art does not disclose at least storing first content responsive to the request in a cache, associating first metadata with the first content, wherein the first metadata is determined by evaluating a parameter of the request, regenerating the request based on the first metadata associated with the responsive content, obtaining second content responsive to the request or replacing the first content with the second content in the cache. Applicant therefore respectfully requests the full allowance of Claims 36-38.

CONCLUSION

Applicant has now made an earnest attempt to place this case in condition for allowance. Other than as explicitly set forth above, this reply does not include an acquiescence to statements, assertions, assumptions, conclusions, or any combination thereof in the Office Action. For the foregoing reasons and for other reasons clearly apparent, Applicant respectfully requests full allowance of Claims 1-38. The Examiner is invited to telephone the undersigned at the number listed below for prompt action in the event any issues remain.

The Director of the U.S. Patent and Trademark Office is hereby authorized to charge Deposit Account No. 50-3183 of Sprinkle IP Law Group in the amount of $350.00 for the payment of new claims added. If any additional fees are necessary, the Director is authorized to charge the deposit account named above.

Respectfully submitted,

Sprinkle IP Law Group
Attorneys for Applicant

Ari G. Akmal
Reg. No. 51,388

Date: April 03, 2006

1301 W. 25th Street, Suite 408
Austin, TX 78705
Tel. (512) 637-9220
Fax. (512) 371-9088